

## 계층신경망을 이용한 다중고장진단 기법 (Multiple fault diagnosis method by using HANN)

이석희(Lee,S.H)\*, 배용환(Bae,Y.H)\*\*, 배태용(Bae,T.Y)\*\*, 최홍태(Choi, H.T.)\*\*

\* 부산대학교 생산기계공학과

\*\* 부산대학교 생산기계공학과 대학원

### Abstract

This paper describes multiple fault diagnosis method in complex system with hierarchical structure. Complex system is divided into subsystem, item, component. For diagnosing this hierarchical complex system, it is necessary to implement special neural network. We introduced to Hierarchical Artificial Neural Network(HANN) for this purpose. HANN consists of four level neural network, first level for symptom classification, second level for item fault diagnosis, third level for component symptom classification, forth level for component fault diagnosis. Each network is multi layer perceptron with 7 inputs, 30 hidden node and 7 outputs trained by backpropagation. UNIX IPC(Inter Process Communication) is used for implementing HANN with multitasking and message transfer between processes in SUN workstation. We tested HANN in reactor system.

### 1. 서 론

프로세스 유니트와 제어시스템의 물리적인 조건변화나 외관조건들은 일반적으로 고장이라 부르는 그 무엇을 야기한다. 정상 작동하에서 하나의 프로세스가 작동할 때, 프로세스 파라메터는 그들의 정상상태에 있어야 한다. 만약 장비에 어떤 물리적인 변화가 정상상태로부터 어떤 편차를 야기하면 프로세스의 모델 파라메터들은 그들의 정상값으로부터 벗어날 것이다. 고장상태를 진단하기 위하여 측정값을 사용하기 보다는 차라리 고장을 분리해 내기 위하여 파라메터를 이용하는 것이 좋다. 프로세스의 파라메터와 측정값 사이에는 관계가 존재한다. 하나의 고장은 프로세스 응답의 일련의 측정과 그들을 파라메터값들로 변역함으로써 진단할 수 있다. 고장진단의 문제는 패턴인식의 하나로 취급될 수 있는데, 프로세스 변수들의 일련 측정에 의하여 고장들을 어떻게 올바르게 측정할 수 있는가 하는 것이다. 다른 말로 하면 Fig.1처럼 측정공간(measurement space)로부터 고장공간(fault space)으로 매핑할 수 있는가 하는 것이다. Watanabe 등은<sup>(1)</sup> 인공신경망을 이용하여 플랜트의 단일고장을 성공적으로 진단하였다. 여기서 우리는 Watanabe와 Himmelblau에 의하여 제시된 모델베이스 고장진단과 같은 다른 기법들 보다 훨씬 빠르고 효과적인 초기의 다중고장진단의 일반적인 문제 해결에 관심이 집중된다. 본 연구에서는 계층화된 시스템의 다중고장진단 기법으로써 먼저 시스템에 고장이 발생되면 시스템 전체에 나타나는 정후를 통하여 부속시스템을 진단하고, 후보부속센중 하나를 취하여 그 하부요소인 아이템과 부속품을 진단하는 것이 목적이다. 이러한 목적을 위하여 본 연구는 다중고장진단을 위한 4레벨의 계층형인 공신경망(HANN)을 이용한다. 상위 2레벨은 서브시스템 고장진단을 통한 고장난 아이템을 구별하는 것이고, 하부 2레벨은 그 아이템속의 고장난 부속품의 고장을 구별해 내는 것이다.

### 2. 고장진단의 정의

진단은 고장의 위치를 찾아가는 과정이며<sup>(2)</sup>, 또한 탐지된 고장을 위한 대응조치를 제시하는 것이다. 진단프로세스(diagnosis process)는 고장탐지(fault detection), 고장위치추정(fault localization), 고장분리(fault isolation)로 요약될 수 있다. 고장탐지는 시스템이 적절하게 작동하지 않을 때 나타나는 정후를 관찰하는 것이다. 센서(sensors)는 시스템의 각종 정후를 표시해 주는 역할을 한다. 고장위치추정은 구별 가능하고, 가장 적절한 원인을 찾아낼 때까지 가능한 원인들을 계속 추출하는 과정이다. 고장분리는 정후소스(symptom source)를 결정하기 위한 탐색과정이다.<sup>(3)</sup>

고장(fault)은 의도한 목적의 수행을 불가능하게 만드는 특성치의 허용되지 않는 편차로 이해될 수 있다. 프로세스 고장이 나타난다면 이것은 가능한 한 빨리 탐지되어야 한다. 이것은 특정 측정 및 미측정 주정변수가 정상값의 어떤 범위내에 있는지를 체크함으로써 행해진다. 이러한 체크과정을 통과하지 않는다면 이것은 고장 메세지를 발생시킨다. 이러한 기능을 통상 감시(monitoring) 혹은 고장탐지(fault detection)로 부르고, 필요에 따라 고장진단(fault diagnosis)이 뒤따라진다. 다음 단계는 고장평가(fault evaluation)이다. 이것은 고장이 프로세스에 어떻게 영향을 미치는지에 대한 평가를 의미한다. 고장들은 사건/순차 분석이나 고장나무목(fault tree analysis)에 따라 각기 다른 위험등급으로 나누어진다. 고장의 영향이 알려진 다음에는 취해져야 할 행동의 결정(decision)이 형성되고, 고장이 허용 가능하다면 작동은 계속 이루어지고, 조건적으로 가능하다면 작동에 변화가 수행되어야 한다. 그러나 고장이 허용되어질 수 없다면 작동(operation)은 즉각 정지되어야 하고, 고장은 제거되어야 한다.

그러나 현재의 고장진단은 주로 시스템의 센서정보를 컴퓨터 자동고장진단 시스템으로 입력해서 실시간 추론이 가능한 장치들이다. 따라서 padalkar 등은 실시간 고장감지와 진단에 대하여 다음과 같이 정의하고 있다. “고장감지(fault detection)는 고장난 시스템의 동정을 감지하는 프로세스로 정의한다. 고장난 시스템 동정은 예측된 시스템 동정으로부터 어떤 편차로 정의된다. 고장진단(fault diagnosis)은 고장난 시스템 동정의 소스(source)와 원인(cause)을 찾는 과정으로 정의된다. 소스와

원인은 무엇이 다른가? 특정의 벨브가 시스템 고장에 책임이 있다고 얘기되어질 수 있다. 이 경우 벨브는 고장난 시스템 동정의 원인이고 소스이다. 사실 좀더 깊이 관찰해 보면 벨브에서 문제의 소스와 원인은 설계에러, 부식, 오동작 등으로 나누어질 수 있다. 이와 같은 분석 즉, 원자의 깊이, 분자의 깊이, 플랜트 조작자의 심리상태까지 임의의 깊이로 행하여진다는 것은 명백하다. 고장시스템동정의 소스(source)는 고장난 시스템 동정에 전적으로 대응하는 시스템의 일부분인 구별가능한 물리적 부품요소로 정의한다. 그리고 고장난 시스템 동정의 원인(cause)은 부식, 오동작, 새가 엔진에 부딪히는 것과 같은 외부객체, 소스 물리적 요소를 고장으로 전개시키도록 기인한 것들로 정의한다.”<sup>(4)</sup> 기계감독의 목표는 정상작동을 통하여 프로세스 변화와 고장들의 탐지와 다른 사건과 프로세스에 손상을 주는 것을 피하는 행동을 취하는 것이다. 이와 같은 업무를 수행하기 위하여 감시부프는 고장탐지와 진단, 고장평가, 작동여부결정, 고장제거등의 기능들을 포함하여 구현된다. 일반적으로 고장진단기법에서 가장 많이 사용되는 프로세스 파라메터에 기초한 고장진단을 위한 체계적인 수준이 Fig. 2에 보여지듯이 세가지 단계로 구분된다.<sup>(5)</sup>

### 1. 데이터 처리 (data processing)

측정된 신호는 진단과 고장탐지를 위하여 적합한 정보로 축소하기 위하여 필터링, 추정등의 여러가지 신호처리 기법들에 의하여 처리된다. 예를 들면, 축소된 정보는 프로세스 모델들이 적용된다면 필터링된 신호요소, 상관관계함수, 혹은 매개변수나 상태변수 추정치 속에 포함되어 있다.

### 2. 고장탐지 (fault detection)

축소된 공정정보에 기초하여 공정속에서 고장탐지를 할 수 있게 하는 특징들이 추출된다. 다음에 이러한 특징의 변화가 정상상태에 대한 참조값으로 결정된다. 이러한 변화들은 고장의 발생과 그 발생 시간을 인식하기 위하여 사용된다. 이러한 임무를 위하여 통계적 결정방법론이 사용된다.

### 3. fault diagnosis (고장진단)

고장사건이 감지된 후 특정들과 그들의 변화들이 고장형태, 고장위치, 고장크기, 고장원인을 결정하기 위하여 분류과정에 입력된다.

### 3. 복합시스템의 계층화 표현

#### 3.1 Structural and functional hierarchy of an complex system

복합시스템은 여러 가지 물리적인 형상에 의하여 운용되는 몇몇 부속시스템들로 이루어져 있다. 모든 부속시스템들은 그들의 기능을 올바르게 수행해야 하고 전체가 조화있게 작동될 수 있도록 각각 서로간에 협의된 동작을 수행한다. 같은 방법으로 부속시스템들의 모든 부품과 요소들은 그들 자신의 기능을 올바르게 수행해야 하고, 부속시스템이 조화있게 작동할 수 있도록 상호간에 단계적으로 수행되어야 한다. 예를 들면, 엔진은 Fig.3에 보인 것처럼 계층의 첫 번째 레벨에서는 엔진 그 자체이다.<sup>(6)</sup> 두 번째 레벨은 엔진을 구성하는 모든 부속시스템, 엔진블록, 크랭크 사프트, 컨넥팅로드, 퍼스톤 어셈블리, 윤활

부속시스템, 냉각부속시스템, 연료, 유도부속시스템, 점화부속시스템, 스타트부속시스템 등이다. 세 번째 레벨은 이러한 부속시스템을 구성하는 부품들 즉 점화부속시스템의 경우 배분기, 스파크 플리그 등이다. 사실 시스템의 신뢰성은 그들의 주요 요소들에 의존한다. 그래서 이와 같은 부품들은 엔진의 구조적 계층을 구축하는데 고려된다. Fig.4에 보이는 것처럼 엔진기능계층에서 첫 번째 레벨은 엔진의 모든 부속시스템이 엔진이 올바르게 작동될 수 있도록 그들의 기능들을 어떤 방식으로 수행해야 하는지를 나타낸다. 두 번째 레벨에서는 몇 가지 그룹들로 이루어져 있는데, 부속시스템의 모든 부품들이 올바르게 작동하기 위하여 그들의 기능을 어떻게 수행해야 하는지를 나타낸다. 한 시스템의 기능계층은 그들의 내부의 기능관계를 나타낸다. 그래서 시스템 기능에서 어떠한 기능도 하지 않는 부품들은 기능계층에서 제외된다. 이와 같이 기능적인 분석에 바탕을 둔 기능계층은 고장의 원인을 찾는데 많은 정보를 제공한다.

### 3.2 계층고장모델 (Hierarchical Fault Model)

진단추론 알고리즘에 적절한 정보는 계층구조모델과 계층기능모델, 그리고 이 두가지 사이의 상호작용에서 발견되어진다. 어떤 사실을 단순화하기 위하여 모든 적절한 고장진단정보를 포함하는 요약된 통합모델이 정의된다. 이 새로운 모델이 계층고장모델(HFM)이다.<sup>(7)</sup> 모든 정보는 이 모델 속에서 자체 규정되어지거나 혹은 다른 모델에서 획득되어진다. 이와 같은 모델을 만드는 이유는 역시 진단추론기법의 이해를 높이는 것과 고장모델링기법표현을 단순화하기 위한 것이다. 고장모델로써 불리워지는 내용들은 계층기능모델에서 각 프로세스를 위하여 정의되어진다. 계층기능모델에서 어떤 프로세스의 고장모델은 프로세스의 고장전과 viewpoint 정보, 고장정보셀 즉, 그들의 모든 부속시스템의 모든 고장모드의 알람 연결들, 요소들의 고장난 상태들과 모든 고장모드를 사이의 고장전과 링크셀과 모든 부속셀의 고장모드들에 대한 정보를 포함한다. 프로세스의 고장전과 모델이 Fig. 5에 보여진다. 계층기능모델에서 모든 프로세스들의 고장모델들의 셀은 계층고장모델(HFM)이다.

### 4. 고장진단기법

고장진단을 위하여 제안된 과정은 다음의 두 가지 가정하에서 논의된다.

1. 고장의 결과는 그들의 정상값으로부터 시스템 파라메터들의 작은 편차로써 나타난다.
2. 모든 고장들은 고장변수를 속에서 대략 같은 상대적인 편차로 발생한다.

Fig.1에서 고장공간에 AND set에 대응하는 다중고장을 고려하자. 예를 들면, 모든 이중, 삼중고장에 대하여 AND set은

$$\{(F_1, F_2), (F_1, F_3), \dots, (F_{n-1}, F_n), (F_1, F_2, F_3), \dots, (F_{n-2}, F_{n-1}, F_n)\}$$

고장을 진단하기 위하여 적절한 신경망이 주어진다면 진단에서의 성공도는 망을 학습하는데 사용된 데이터의 특성에 의존한다. 망이 단일고장을 진단하기 위하여 학습되어졌다면 망이 다중고장에 대한 결정을 내리기 위하여 어떻게 사용되는지를 알아보자. 앞에서 언급된 것처럼 측정값  $y$ 는 프로세스에서 진

단되는 파라메터  $p$ 의 함수로써 표현될 수 있다.

$$y = f(p) \dots (1)$$

가정 1로 부터

$$p = p^0 + \Delta p \dots (2)$$

여기서  $p^0$ 는 정상작동조건을 위한 파라메터들의 벡터이다. 즉 정값  $y$ 는 식(2)를 식(1)에 대입함으로써 선형방정식의 셀으로서 출력된다.

$$y = y_0 - \Delta y = f(p^0) - A\Delta p \dots (3)$$

여기서

$$A = \left[ \frac{\partial f(p)}{\partial p} \right]_{p=p^0} = [ a_1, a_2, a_3, \dots, a_n ]$$

은 상수계수를 가진 Jacobian matrix이다.  $y_0 = f(p^0)$ 이기 때문에 다음과 같은 선형방정식으로 표현가능하다.

$$\Delta y = a_1 \Delta p_1 + \dots + a_i \Delta p_i + \dots + a_n \Delta p_n \dots (4)$$

우리는 위식을 단일고장을 위한 학습데이터에 기초한 다중고장을 위한 분류전략을 위한 기초로 사용한다. 진단전략의 구체적인 예시를 Fig.1에서 보인 고장공간에 속하는 AND set으로, 각기 다른 고장진단을 위한 측정에서 예상되는 출력편차의 세 가지 예는 다음과 같다.

#### Case 1 : single faults

AND set = { F<sub>1</sub>, F<sub>2</sub>, ..., F<sub>n</sub> }

Fault	Deviation
F <sub>1</sub>	$\Delta y_1 = a_1 \Delta p_1$
F <sub>2</sub>	$\Delta y_2 = a_2 \Delta p_2$
.	.
F <sub>n</sub>	$\Delta y_n = a_n \Delta p_n$

#### Case 2 : all of the double faults

AND set = {(F<sub>1</sub>, F<sub>2</sub>), (F<sub>1</sub>, F<sub>3</sub>), ..., (F<sub>n-1</sub>, F<sub>n</sub>)}

Fault	Deviation
F <sub>1</sub> , F <sub>2</sub>	$\Delta y_1 + \Delta y_2$
F <sub>1</sub> , F <sub>3</sub>	$\Delta y_1 + \Delta y_3$
.	.
F <sub>n-1</sub> , F <sub>n</sub>	$\Delta y_{n-1} + \Delta y_n$

#### Case 3 : double and triple faults which include F<sub>1</sub>

AND set = {(F<sub>1</sub>, F<sub>2</sub>), ..., (F<sub>1</sub>, F<sub>n</sub>), (F<sub>1</sub>, F<sub>2</sub>, F<sub>3</sub>), ..., (F<sub>n-1</sub>, F<sub>n</sub>)}

Fault	Deviation
F <sub>1</sub> , F <sub>2</sub>	$\Delta y_1 + \Delta y_2$
F <sub>1</sub> , F <sub>n</sub>	$\Delta y_1 + \Delta y_n$
F <sub>1</sub> , F <sub>2</sub> , F <sub>3</sub>	$\Delta y_1 + \Delta y_2 + \Delta y_3$
.	.

F<sub>1</sub>, F<sub>n-1</sub>, F<sub>n</sub>  $\Delta y_1 + \Delta y_{n-1} + \Delta y_n$

#### 5. Hierarchical Artificial Neural Networks

본 연구에서 계층신경망(HANN)을 이용한 고장진단을 다중처리화하기 위하여 SUN workstation상에서 4개 레벨의 64개의 신경망이 동기화를 이루어 돌아가는데 여기서 각 신경망의 weight값은 off-line으로 학습되어, 신경망의 특성값과 함께 파일에 저장되어 있다. 일단 진단이 시작되면 프로세스는 상위 두 개의 계층(first level, second level)의 신경망이 수행되고, first level에 서브시스템레벨 입력데이터를 읽어 들여 결과를 출력하면, 그 고장결과수 만큼 second level이 활성화되고, message queue를 통하여 입력데이터를 넘겨줌과 동시에 OR process에게도 넘겨진다. OR 프로세스는 넘겨받은 결과값을 바탕으로 메세지큐 번호를 찾아다니면서 second level의 결과를 전달받아 OR연산한 후 하부 component 고장진단을 위하여 하위계층(third level)을 활성화시키는데, 이때 활성화되는 신경망의 수는 second level의 고장갯수이다. 아이템레벨 입력데이터를 읽어들여 결과를 OR 프로세스로 넘긴후 third level은 forth level 7개 신경망을 활성화시킨 후 결과값을 출력하면, OR프로세스는 third level의 결과에 해당하는 forth level의 결과를 OR연산한다. 이렇게 함으로써 계층신경망은 고장아이템과 고장component를 진단한다. Fig.1의 고장공간속에서 단일, 이중 고장진단의 모든 경우를 고려하면

$$\{ (F_1, F_2, F_3), \dots, (F_n, (F_1, F_2), \dots, (F_{n-1}, F_n)) \}$$

이중고장의 수  $n_a$  그리고 전체 고장수  $n_t$ 로 주어진다.

$$\begin{aligned} n_t &= 1 + [n + (n-1) + \dots + 2 + 1] = 1 + [n(n+1) \\ n_d &= (n-1) + \dots + 2 + 1 = n(n-1)/2 \end{aligned}$$

첫 번째 레벨의 단일신경망은 전체 고장셀로 정상상태의 경우를 포함한 고장의 수  $1 + [n(n+1)/2]$ 개의 학습패턴을 학습해야 한다. 첫 번째 레벨의 출력 노드에서 단일고장의 경우는 출력값이 1인 것이 한 개가 있고, 이중고장의 경우는 두 개가 있다. Fig.6를 살펴 보면, Net<sup>0</sup>으로 붙어진 첫 번째 레벨에서 모든 측정값이 입력으로 사용된다. 다시 말해서 i 번째 출력 노드가 첫 번째 레벨에서 활성화되면 두 번째 레벨에서 i 번째 Network인 Net<sup>i</sup>의 출력이 OR operation에서 사용되는 셀으로 형성된다. OR 프로세스는 두 번째 레벨에서 결과를 받아 세

번째 레벨의 신경망이 고장 결과에 해당되는 신경망을 활성화 한다. 예를 들면, 고장( $F_5F_6F_7$ )이 일어날 때 첫 번째 레벨의 출력은  $\{F_5F_7\}$ 이 되고, 두 번째 레벨에 있는 네트워크  $Net^5$ 와  $Net^7$ 이 활성화된다.  $Net^5$ 은  $\{F_5F_6F_7\}$ 을,  $Net^7$ 은  $\{F_5F_7\}$ 을 출력한다.  $\{F_5F_6F_7\}$ 과  $\{F_5F_7\}$ 은 OR 프로세스로 보내지고, OR 연산되어,  $\{F_5F_6F_7\}$ 이라는 결과가 산출된다. 이 결과가 HANN의 출력이 된다.

## 5. 계층형신경망의 적용예

Fig. 7은 헬탄을 톨루엔으로 반응시켜 새로운 물질을 합성하는 PI feedback controlled reactor를 나타낸 것으로 거의 정상상태 조건하에서 작동한다. 아주 극단적인 돌발적인 사고는 일어나지 않는다고 가정한다. 또한, 초기 단계에서 포함된 프로세스 고장은 강하게 억제되고 PI 제어기에 의하여 완화되도록 한다. 링크 속에 저장된 헬탄은 프로세스 펌프 1을 통하여 반응으로 보내지고, 반응로 속에 일어나는 작용으로서 반응비율은 반응로 속의 온도에 의하여 조절된다. 증기는 재생펌프 2를 통하여 열교환기로 공급된다. 열교환기의 출구값은 PI 제어기에 의하여 제어되는데, 제어되는 변수는 반응온도이다. 여기서 열교환기의 제어는 전자제어회로나 컴퓨터를 통하여 이루어진다고 가정한다. 그리고, 여기서는 작동유체의 상태에만 관심이 있으므로, 제어기에서의 고장은 포함시키지 않는다. HANN의 적용예로써 7가지 가능한 고장예를 고려하면 다음과 같다.

$$\begin{aligned} F_1 &= 0.1\% \text{ 감소 } \text{명령 신호 } u_0 \\ F_2 &= 2.0\% \text{ 감소 } \text{주파수 } k_o \\ F_3 &= 0.1\% \text{ 감소 } \text{활성에너지 } E_a \\ F_4 &= 2.0\% \text{ 감소 } \text{열교환기 개인 } K \\ F_5 &= 2.0\% \text{ 감소 } \text{전체 열전달계수 } h \\ F_6 &= 1.0\% \text{ 감소 } \text{입구에서의 헬탄} \\ F_7 &= 1.0\% \text{ 감소 } \text{출구에서의 증기 반응온도 } T_i \end{aligned}$$

파라미터속에서 이와같은 변화들은 같은 정도(2.0%)의 프로세스 변수속에 오차를 유발한다. 고장공간은 7차원( $n=7$ )이고, 상관관계 분석에 의해서 측정 공간 속에서 고장진단에 필요한 변수는 다음과 같이 7개만으로 가능해진다.

$$\begin{aligned} y_1 &= \text{열교환기 출구 온도 } T_h \\ y_2 &= \text{반응기의 출구 온도 } T \\ y_3 &= \text{PI 제어기의 출력 } s_i \\ y_4 &= \text{반응기 출구 물질 농도 } 1 \\ y_5 &= \text{반응기 출구 물질 농도 } 2 \\ y_7 &= \text{반응기 입구 물질 농도 } \\ y_9 &= \text{반응기 입구 온도 } T_i \end{aligned}$$

본 연구에서는 최소의 측정으로 최소의 오분류를 가진 고장을 진단하는 것이다. 처음에는 문제속에 포함된 모든 변수를 전부 선택했다. 그러나 측정의 전처리에 의하여 네트워크의 입력을 최소로 할 필요가 있다. 측정을 적당하게 조절하기 위하여 프로세스 측정값의 모든 오차  $\Delta y_i$ 는 측정값의 2%로 하여 정규화된다. 변화계수는 2%이기 때문에 몇몇 노이즈가 더해진다면 분류는 올바르게 수행된다. Table 1에서 학습테이터는 프로세스 측정의 정규화와 데이터 축소(reduction)에 의하여 얻어진다. 이와 같은 테이터는  $Net^0$ 를 학습하는데 사용된다. 고장  $F_1$ 이 존재할 때, 출력은  $\{1 0 0 0 0 0 0\}$ 이다. 7개의 고장에 대응한 7개의 네트워크  $Net^i(i=1,2,3,\dots,7)$ 가 두 번째 레벨

에 존재한다.  $Net^i$ 를 위한 학습테이터는 앞에서 설명했듯이 두 개의 고장을 위하여 Table 1에 더함으로써 얻을 수 있다. Table 2에서  $F_1$ 을 포함하는 모든 이중고장의 경우를 조사하자. 예를들면  $\{F_1, F_2\}$ 를 위한 학습테이터는 Table 1에서 2행 ( $F_1$ 에 대응하는 고장)테이터를 3행( $F_2$ 에 대응고장)에 더함으로써 발생되고 출력은  $\{1 1 0 0 0 0 0\}$ 이 된다. 본 연구에서 사용된 신경망의 특성은 각 계층별 신경망의 입력은 7개, 은닉층은 30개, 출력층은 7개이고,  $\eta = 0.1$ ,  $\alpha = 0.9$ 로 세팅하고, 반복은 최소에러가 0.008보다 적을때 까지 계속된다. 요구되는 반복수 평균은 6800이고, 학습패턴의 수는 29개였다. (normal case 1, single faults 7, double faults 21) 비록 학습은 단일, 이중 고장에 기초하고 있지만, 테스트는 단일, 이중, 삼중고장들에 의하여 수행된다. 테스트할 패턴수는 정상 1개, 단일 7개, 이중 21개, 삼중 35개로 총 64개이다. output layer 속에 first level의 노드출력이 0.5보다 클 때 정후가 존재하는 것으로 고려하고, second level에서 출력이 0.9보다 큰 경우에 아이템 고장이 존재하는 것으로 하고, third level에서는 노드출력이 0.5이상이면 component속에 고장정후가 존재하고, forth level에서는 출력노드가 0.9 이상이면 component에 고장이 존재하는 것으로 한다. Tables 3,4는 단일, 이중, 삼중고장을 위한 진단 결과를 보여준다. 테이블로 부터 알수 있는 것은 다음과 같다. Fig. 8은 HANN의 구현예를 나타낸 것이다.

## 6. 결 론

본 연구에서는 다중고장진단하기 위하여 사용될 수 있는 UNIX 운영체제에서 멀티테스킹으로 IPC(Inter Process Communication)서비스를 이용한 계층형신경망(HANN)이라 불리는 neural network가 제시되었다. HANN을 반응로의 고장진단에 적용한 결과 다음과 같은 결론을 내린다.

1. HANN은 기존의 신경망이 단일, 이중, 삼중고장에 대한 데이터를 전부 학습하는 반면에 본 연구에서 제시된 신경망은 각각에 맞는 고장을 선별하여 학습함으로써 다중고장 인식률을 높일 수 있다.
2. HANN은 인체의 신경망을 계층화하여 표현함으로써 각 프로세스는 독립적으로 학습될 수 있고, 다음에 수정시 필요한 부분만 수정함으로써 수정이 가능하다.
3. 본 HANN을 전문가시스템이나 다른 모델베이스 고장진단과 연결 운용이 가능할 것으로 사료된다.

## 참고문헌

- (1) Watanabe, K. et al., " Incipient Fault Diagnosis of Nonlinear Processes with Multiple Causes of Faults", Chem. Eng., Sci., vol.39, pp.491 , 1983.
- (2) Khaksari, M., " Expert diagnostics system ", First Int. Conf. Industrial and Engineering Applications of AI and Expert Systems, Tullahoma, pp.43-53, 1988
- (3) Iserman, R., " Process Fault Detection based on Modeling and Estimation Method - A Survey ", Automatica, Vol.20, pp.387-394, 1984.
- (4) Padalkar, S., " Graph Based Real-Time Fault Diagnostics",

Proceeding of the 4th Conference on AI for Space Applications, pp.115-123, 1988

- (5) Iserman, R., "Process Fault Diagnosis based on Dynamic Models And Parameter Estimation Methods", Fault Diagnosis in Dynamic Systems : Theory and Applications, Prentice Hall, pp.253-291, 1989.
- (6) Zheng, X., Yang, Z. and Shi, H., "A Knowledge-Based Diagnosis System Automobile Engines", International Journal of Advanced Manufacturing Technology, Vol.3, No.3, 1988.
- (7) Narayanan, N., "A methodology for knowledge acquisition and reasoning in failure analysis of systems", IEEE Transaction on Systems, Man and Cybernetics, Vol.17, No.2, pp.274-pp.288, 1987.
- (8) Bach, M.J., The Design of the UNIX Operating System, Prentice Hall, pp.105-237, 1986.
- (9) Stevens, W.R., Advanced Programming in the Unix Environment, Addison Wesley, pp.188-212, 1992.
- (10) 흥룡과학출판사 편집부, UNIX 시스템 프로그래밍, 흥룡과학출판사, 1991.
- (11) 윤유현, 백정현, UNIX 시스템 프로그래밍, 정의사, 1995

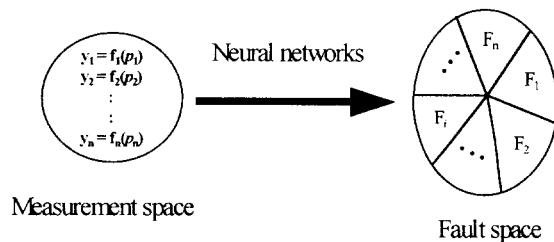


Fig. 1 Mapping from a measurement space to a fault space

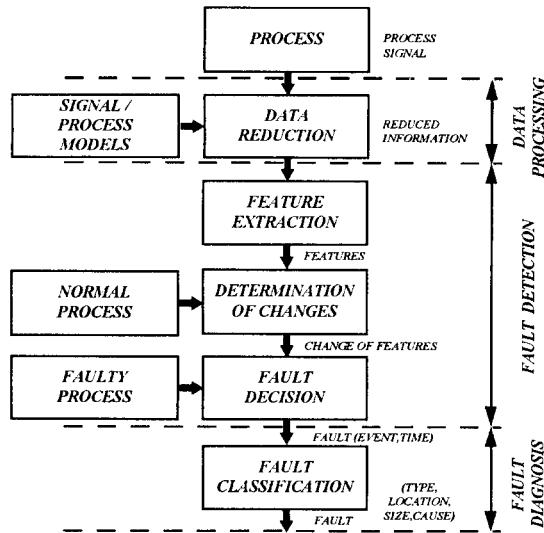


Fig. 2 Fault diagnosis procedure

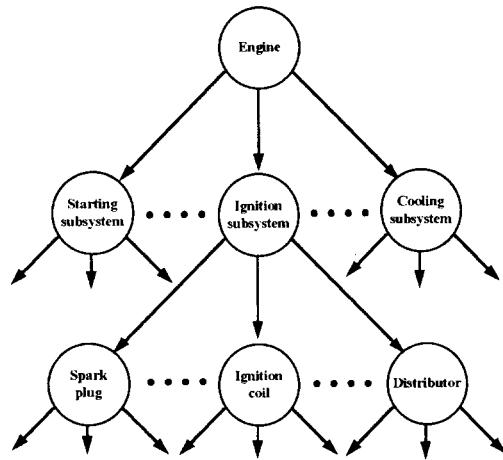
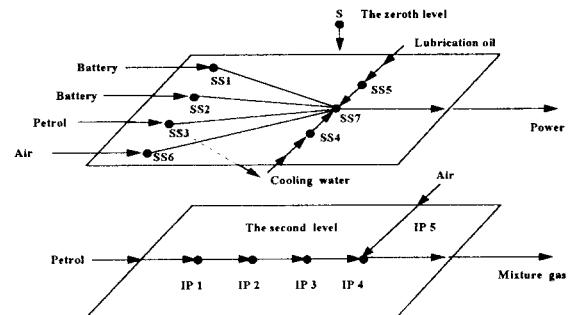


Fig. 3 Structural hierarchy of engine



S - engine, SS1 - starting subsystem, SS2 - ignition subsystem  
SS3 - fuel subsystem, SS4 - cooling subsystem, SS5 - lubrication subsystem  
SS7 - valve train, SS7 - engine body, IP 1 - fuel tank, IP 2 - fuel filter  
IP 3 - fuel pump, IP 4 - catalytic converter, IP 5 - air filter

Fig. 4 Functional hierarchy of an engine

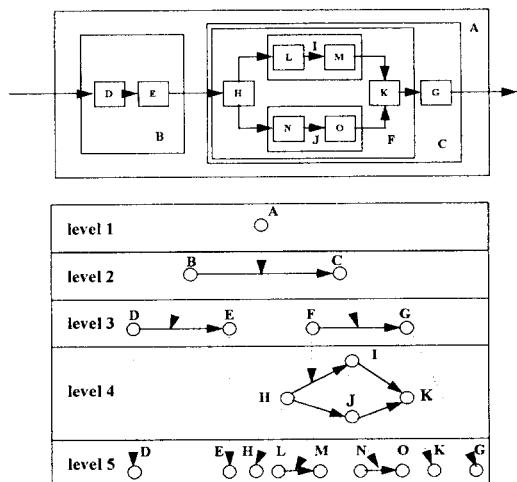


Fig. 5 Hierarchical fault propagation model

